MATES Computer Science

Senior Capstone Project Final Report

Project Title	Bookish
Team Members	Malia Porter & Caleigh Wlazlowski
Link to GitHub	https://github.com/caleighwlaz/Bookish

I. Executive Summary

We have created a website called Bookish, which is similar to Goodreads. Goodreads is a social cataloging website and a subsidiary of Amazon where readers can search its database of books, annotations, quotes, and reviews and sign up and register books to generate library catalogs and reading lists. Our book website would allow readers to log in, create a profile, set reading goals, and add books to their reading list. Users could also keep track of their reading activity and discover new books to read. The site currently includes a homepage (where there is a summary of our website and where future book and reader leaderboards would be located), an About Us Page, a Login Page, a Sign-Up Page, and an All Books Page.

II. Detailed Summary

Overview

Our project was chosen because we both have a love for books. Our goal was to create a website for readers to track their reading history, view book and user leaderboards, and possibly get recommendations. The site would allow readers to set goals to reach and add books to their reading list. Users will also be able to keep track of their reading activity and discover new books to read. Through the front-end, Malia used HTML, CSS, and JavaScript to create the website and the design elements. Through the back-end, Caleigh used Python, JavaScript, and MySQL to clean and

upload the book dataset and manage databases. The concept of our website was similar to Goodreads with additional features such as book and user leaderboards.

Technologies and Versions

We used **Visual Studio Code** (VS Code) to code this website. The languages we used to program the site were **HyperText Markup Language** (HTML), **Cascading Styles Sheets** (CSS), **Sassy Cascading Style Sheets** (SCSS), **JavaScript,** and **Python**. Caleigh also used **Microsoft Excel** for the book dataset and attempted to use **SQLite**, **PostgreSQL**, and **MySQL Workbench**.

Visual Studio Code was connected to our Github and used to edit our .html files, .css files, .css.map files, .scss files, .md files, .csv files, and .sql files.

Key Milestones

The key milestones within the first two weeks, from February 12 to February 23, consisted of learning the languages necessary to complete the project. Malia completed multiple courses on Codecademy for front-end development and Caleigh completed some courses on Codecademy for front-end development and multiple for back-end development. We worked together to pick out the color palette for the website and design the homepage in Google Drawings. Finally, Caleigh found two datasets to use and started cleaning the data while Malia began coding the homepage in VS Code.

During the next progress period, from February 26 to March 8, we both continued to complete courses on Codecademy. Caleigh found book datasets and uploaded book covers to VS Code. Malia finished coding the layout of the homepage using CSS Flexbox and Grid.

From March 11 to March 22, Caleigh finished uploading the book covers to GitHub, continued working on the “Back-End Engineer” career path on Codecademy, and uploaded the book covers dataset into Python and Jupyter Notebook files. Malia designed and implemented the favicon for the website, designed the Sign-Up and Login pages in Google Drawings, and installed Live Sass Compiler to start using SCSS.

During the next progress period, from April 8 to April 19, Caleigh researched SQL database to use, found a way to convert the Excel file into a SQL database on SQLite, started creating a sample book database in Postgres, and continued working on the “Back-End Engineer” career path. Malia created the hamburger menu and added a media query to hide the normal navigation links and show the hamburger menu on smaller screens.

During the next two weeks, from April 22 to May 3, Caleigh added ten books from each genre into a test table in PostgreSQL, started to filter the main dataset, and switched from PostgreSQL to MySQL. Malia modified the sidebar and nearly finished the front-end for the About Us Page and Login Page.

Over the next two weeks, from May 6 to May 17, Caleigh finished the test book database on MySQL, continued to filter the main dataset, started to work on the All Books Page, and continued working on the "Back-End Engineer" career path. Malia finished the About Us Page and the front-end for the Sign-Up and Login pages.

Scope Changes

Originally, we had planned to have login functionality and to create spaces for users to set reading goals and view active leaderboards. Unfortunately, this project was much more complicated than anticipated. There was quite a learning curve for the front-end that hindered progress and there were a lot of errors for the back-end. We also had reach goals of connecting users with similar characteristics and creating book and author recommendations, but these goals were never a priority. We did manage to create the homepage, About Us Page, Login Page, Sign-Up Page, and the All Books Page for the website.

Next Steps

As for the next steps for continuing this project in the future, we would like to finish the All Books Page and accomplish our original goals. That is, we would like to create a functional login system and an individualized experience for each user so they can track which books they've read/want to read, view statistics on how many books/pages they've read, and get personalized book recommendations. We would also like to add leaderboards to show users with the highest number of books read and books/authors with the highest ratings on the site.

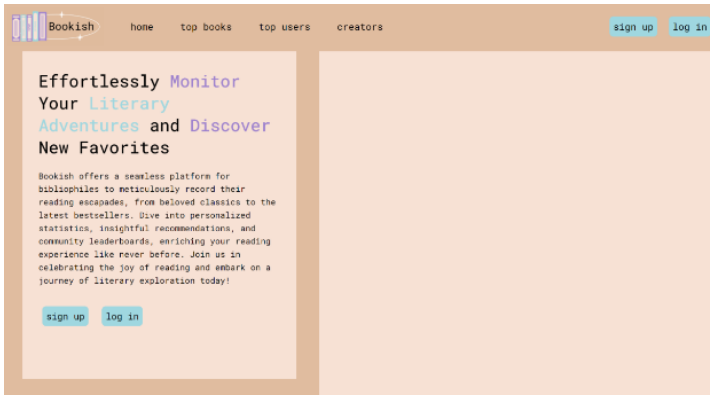
Individual Contributions

The work was divided into front end and back end. Malia focused on the front end of the website while Caleigh concentrated on the back end. There was minimal crossover between front end and back end (like when Caleigh made the "all books" page and used HTML and CSS), but for the most part, we worked separately.

Malia:

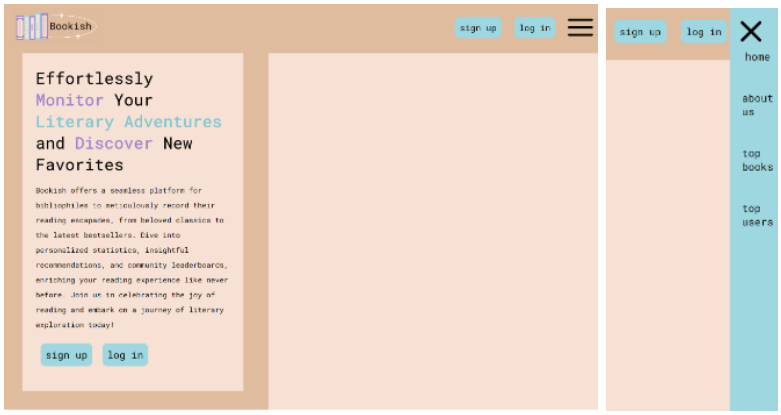
In the first two-week period, I completed multiple lessons on the Codecademy “Build a Website with HTML, CSS, and GitHub Pages” skill path to learn the basics of the languages that I would be using for this project. I also used Google Drawings to help design the homepage of the website and create the website logo. To finish the first progress period, I started coding the homepage in VS Code and realized I needed to learn about CSS grids.

During the second progress period, I completed courses on Codecademy covering CSS layout, positioning, and Flexbox. Then, I implemented what I learned by creating the website grid, positioning the header elements using Flexbox, using relative units (rem and fr) to allow for responsive design, and adding basic transitions to the buttons so they would change from blue to purple on hover. I also installed Live Server in VS Code to view the website more easily.

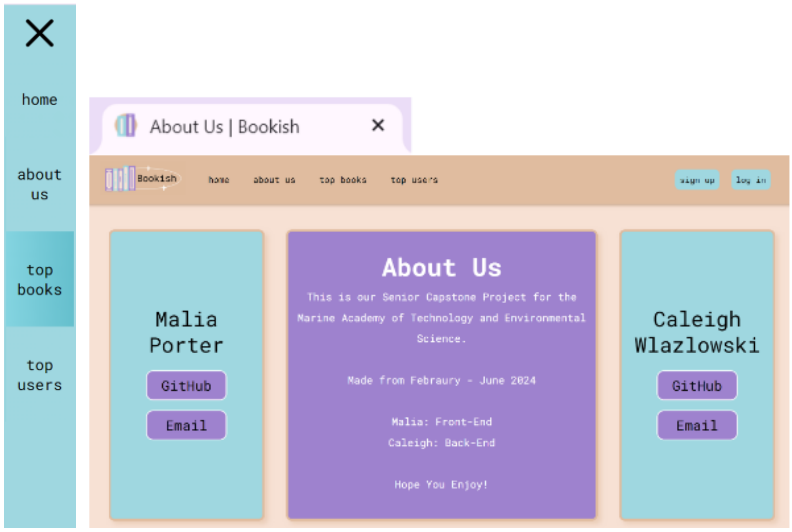


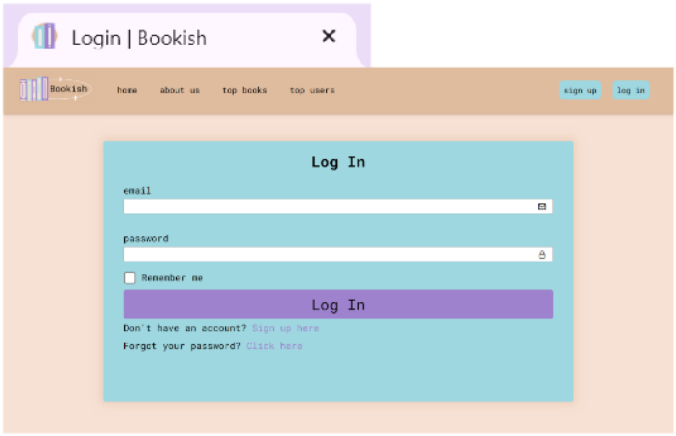
During the third progress period, I created the website’s favicon in Google Drawings. I had to keep testing the favicon on the website and redesigning it because it was very difficult to create an image that looked good on such a small scale. Once I finalized the design for the favicon, I converted it from a .png to a .ico and added it to the head of the website in HTML. Then, I learned about SCSS, installed Live Sass Compiler in VS Code, and started nesting rules and creating color variables in SCSS files. Afterward, I designed the Sign-Up and Login Pages in Google Drawings and set my sights on creating a hamburger menu.

During the fourth progress period, I found a tutorial on YouTube on creating a hamburger menu with only HTML and SCSS. I followed along with the tutorial, set overflow-x to hidden on the body to hide the sidebar, and positioned the elements in the sidebar using Flexbox. After viewing how the page looked at different widths using DevTools, I added a media query to show the hamburger menu and hide the normal navigation when the screen size is less than 1240 pixels.

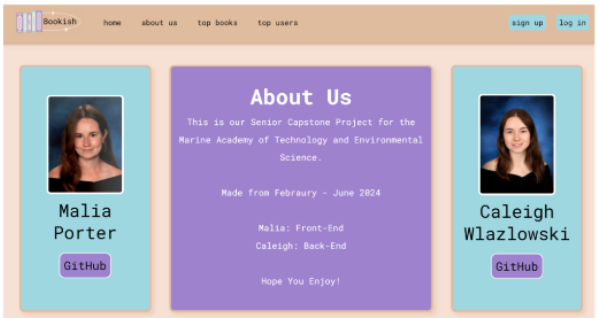


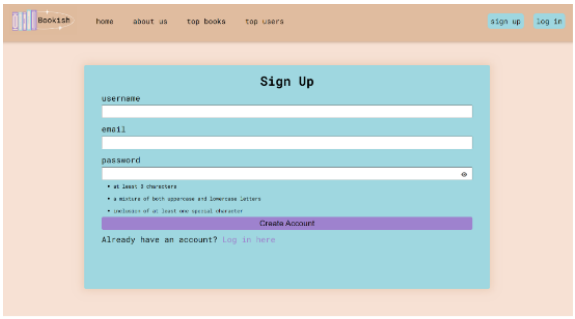
During the fifth progress period, I first fixed the sidebar and hamburger menu. I added a media query so the sidebar would disappear on larger screens, added a slide-in background effect to the links on hover, and centered the links in the sidebar. I set the sidebar's positioning to fixed so it would take up the full screen height and set the hamburger menu's positioning to fixed when the sidebar was open (so that users could still close the sidebar when they scrolled down) and absolute when the sidebar was closed (so that it would return to its original position when a user closed the sidebar while they were further down the page). Next, I started working on the About Us Page. I created and positioned the three cards, added shadows, typed the description for the project, and added links for our GitHub accounts. Then, I started working on the Login Page. I added icons to the email and password inputs (which would later be discarded) and changed the "Remember me" checkbox color to purple when clicked. Finally, I changed the top background color to purple on phones, changed the buttons to be links that looked like buttons for accessibility, and changed the page titles.



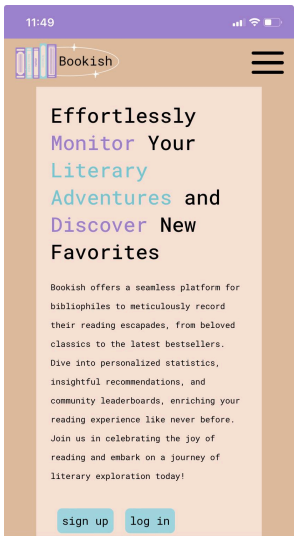


During the sixth progress period, I finished the About Us Page by adding our pictures and fixing the formatting. Then, I finished the Login and Sign-Up pages. I replaced the previous icons with a toggle password icon that allows users to view what they typed in the password input when clicked. When creating the toggle password function, I used JavaScript and Font Awesome icons and made sure to change the cursor to a pointer on hover. I also surrounded the input boxes in divs so the icons wouldn't move around as much when the screen size changed.





To finish up the project, I focused on phone compatibility. I made the sign-up and login buttons on the homepage transfer to the sidebar on small screens, added a media query for the cards in the About Us Page to be stacked on one another, and fixed how the Login and Sign-Up pages look on phones.



Caleigh:

For the first progress period, I completed an HTML course, a CSS course, and a JavaScript course on Codecademy. Also on Codecademy, I started a skill path for building a website with HTML, CSS, and GitHub pages as well as started a “Back-End Engineer” career path. I attempted to find book databases online, but most were not available. I eventually found two book datasets on Kaggle that I cleaned up. I learned how to convert a CSV file into an Excel file as well, using the “Text to Columns” button.

During the second progress period, I continued working on Codecademy’s “Back-End Engineer” career path, which involved setting up a dev environment and reviewing web development fundamentals (HTML, CSS, and JavaScript). I searched for a better book dataset(s) to use because I wanted one that included the book covers. I stumbled upon another dataset on Kaggle that came with over 32,000 book covers in separate files. I also connected Github to VS Code and taught myself how to use it, which I continued learning through errors throughout the rest of the progress periods. In addition, I had problems with connecting the Python interpreter and committing, but those ended up getting fixed during this progress period. I learned that VS Code will not let you commit over 10,000 things at a time, so I had to learn to commit by book covers one piece at a time. Also, I learned that I have to commit and push in order for the new changes to appear on GitHub.

During the third progress period, which was mid-March, I finished uploading the book covers to Github. I used x-lookup to connect all of the datasets I had from before into one dataset, but I had

missing titles and ISBN values. This is how I found out that each book has multiple ISBN values—one for each of its publications. I ended up just working with the dataset that came with the book cover files because it was a more complete file and had the book covers that actually worked. I uploaded the set into Python and Jupyter Notebook files in order to start analyzing general information. I also continued the "Back-End Engineer" career path with more review of web developments and learning to use the command line.

During the fourth progress period, I researched SQL databases to use. I started with SQLite because I did not need admin permission to download it. I downloaded the ESF Database Migration Toolkit - Pro in order to convert the Excel file into a SQL database. Unfortunately, SQLite errored out and crashed before I was able to properly use it. I then attempted to use PostgreSQL, which was much better. I downloaded Postbird in order to access it. I had to learn how to properly use Postgres, like only running one query at a time and then switching to updating rows in the "Content" tab. I started to create a sample book database on Postgres. This was a much smaller section of the book dataset to use on the website and format before uploading the whole dataset. I also continued the "Back-End Engineer" career path with learning JavaScript and database basics.

title	author	isbn	genre	img_paths
This is Going to Hurt	Adam Kay	9780000000000	Medical	dataset/Medical/0000001.jpg
Thinking, Fast and Slow	Daniel Kahneman	9780000000000	Medical	dataset/Medical/0000002.jpg
When Breath Becomes Air	Paul Kalanithi	9780000000000	Medical	dataset/Medical/0000003.jpg
The Happiness Trap	Russ Harris	9780000000000	Medical	dataset/Medical/0000004.jpg

During the fifth progress period, I finished up the sample database on PostgreSQL and tried to research connecting Postgres to an HTML file or being able to convert an Excel file into a SQL database. I did not find anything on how to do it, so I ended up changing to MySQL Workbench. Unfortunately, MySQL needed admin permissions to download, and IT John was not available for most of the progress period, so I focused on cleaning up the dataset. I tried to add all of the different genres for one book into one column, but it ended up being a mess, and it would be more trouble later when trying to upload it. I also started to delete the repeated book covers in order to save room. I got MySQL two days before the progress period ended, so I was able to start making a test book database with fewer rows than Postgres'. I also continued working with databases on the "Back-End Engineer" career path.

During the sixth progress period, I finished the test book database on SQL and continued to filter the main dataset. I updated the back-end branch on Github and learned how to pull and push commits in order to not cause errors. I also started setting up formatting for an “all books” page on an HTML file.

```
    <body>
        <div class="background">
            <div id="grid-container">
                <header>
                    <nav>
                        <a href="index.html">
                            <img src="/images/bookish-logo.png" alt="Bookish Logo" id="logo">
                        </a>
                        <a href="index.html" class="nav-link">home</a>
                        <a href="about-us.html" class="nav-link">about us</a>
                        <a href="#" class="nav-link">top books</a>
                        <a href="#" class="nav-link">top users</a>
                        <!-- <a href="all-books.html" class="nav-link">all books</a> -->
                    </nav>
                    <div class="header-login-area">
                        <a href="sign-up.html" class="sign-up">sign up</a>
                        <a href="login.html" class="login">log in</a>
                    <label class="hamburger-menu">
                        <input type="checkbox">
                    </label>
                    <aside class="sidebar">
                        <nav>
                            <a href="index.html" class="sidebar-link">home</a>
                            <a href="about-us.html" class="sidebar-link">about us</a>
                            <a href="#" class="sidebar-link">top books</a>
                            <a href="#" class="sidebar-link">top users</a>
                            <!-- <a href="all-books.html" class="sidebar-link">all books</a> -->
                        </nav>
                    </aside>
                </div>
                </header>
            </div>
        </div>
    </body>
</html>
```

To finish up the project, I made and connected the .css file, .css.map file, and .scss file to the “all books” page. I also added the “all books” page to the top bar and sidebar throughout the website. In HTML, I typed up and formatted the 33 genres of the book dataset. I tried to convert the Excel file to a SQL database, but I just thought it would be easier to use the Excel dataset just because of timing. I attempted to use JavaScript or Python to connect the Excel dataset to the website, but I kept getting errored out or the website would be stuck loading. I manually wrote in and spaced out the first few books for the first few genres on the page in order to make it look more presentable.

III. Demo Video

The video would not upload to this document, so Caleigh submitted it separately on Google Classroom.

IV. Reflection and Closing Thoughts

Malia:

This was my first time using any of these languages and trying out web design. Although the Codecademy courses were very helpful, I wish that I jumped into the project sooner. It was much easier to learn through application, and I realized that a full understanding of the syntax of the language wasn't essential to get started; I just needed to know the fundamentals and the logic. The homepage took a lot longer than predicted due to the learning curve, but it helped familiarize me with HTML and SCSS and prepared me for the rest of the pages. I'm glad that I designed the pages in Google Drawings before I started coding them so that I had a vision of what I wanted the page to look like before I started coding. I'm happy with the project we picked, even if it turned out to be much more difficult to execute than either of us expected. I'm also satisfied with the technology used, especially SCSS, which made it easy to reuse styles and allowed for the nesting of rules.

I felt that a progress report every two weeks was good because it forced me to stay on task without having to spend too much time writing the reports. I don't have any suggestions for the course next year; I was happy with the current setup. Although I enjoyed learning about web development and am grateful for the opportunity to help create Bookish, I do not plan on continuing development for this project after the course is over.

Caleigh:

This was my first time learning new coding languages by myself and applying them, so there were definitely a lot of hurdles I had to overcome. I was also very forgettable and took more time making notes on the Codecademy courses, so I wouldn't forget what I learned the next day. Unfortunately, a lot of things went wrong on my end because I kept running into errors and issues (just my luck). I was not able to complete many of the goals I had set for myself and spent way too much time trying to figure out a dataset to use. I'm glad that I stumbled into the dataset with 32,000 book covers. If I could go back, I wouldn't change the project. I learned a lot from the many errors that I came across. I also enjoyed using VS Code; however, the connection between GitHub and VS Code was not ideal and was probably the most frustrating part for me.

I believe that the biweekly progress reports are very helpful, and I think that they should be kept. I was hoping to continue developing the website, but I keep running into different GitHub-related errors, so I do not think I have the patience anymore to try and fix them. I don't think any tweaks should be made to the overall orientation of the projects.